Little's Formula

Define:

$L(t) =$ # of customers in the system by time t

$\bar{L}(t) = \frac{1}{\mathrm{t}}\int_0^t L(t)dt:$ AVG # of customers in the system by time t

$N(t) =$ # arrivals by time t

$\bar{\lambda}(t) = \frac{N(t)}{t}:$ AVG arrival rate by time t

$W(t) =$ time in the system (waiting and being serviced) of all customers that arrived by time t

$\bar{W}(t) = \frac{1}{\mathrm{N(t)}}\int_0^t W(t)dt:$ AVG time in the system of all customers that arrived by time t

We have if system starts empty and is empty by time t:

$$\bar{L}(t) = \bar{\lambda}(t)\ \bar{W}(t)$$

As $t \to \infty$, with probability 1:

$\bar{L}(t) \to \bar{L},\ \bar{W}(t) \to \bar{W},\ \bar{\lambda}(t) \to \bar{\lambda},$ and

$$\bar{L} = \bar{\lambda}\bar{W}$$

Corrections on theoretical concepts

- HW 1
  - Run simulation until event list empties
  - Calculation of Avg # customers in system: issues with numerator and denominator
  - Average arrival rate not equal to AVG inter-arrival time
- HW 2
  - Incomplete or unclear list of parameters and state variables
  - Missing time to travel between station in the event graph
  - Missing interarrival time in edge from Run→Arrival events
  - Missing arrival counter necessary to measure percentage of balks
  - Backwards edge going from EOS→SS in system with no buffer
  - Passing parameters across edges
  - Missing explicit formula to find % balks: # balks / # arrivals = (# arrivals - # serviced) / # arrivals
- HW 3
  - Passing parameters across edges
  - Condition on crusher and shovel available before start of service
  - Clear event graph
  - No canceling edges
  - Prioritize large trucks at the crusher
  - Q.add() with empty arguments
  - Passing objects (trucks) across edges

Corrections on Java implementation

- Lab 1:
  - State variables are protected
  - Parameters are private
  - In doArrival: do not increment numberArrivals twice
  - Talk about variable manipulation inside doXxx methods
- Lab 2:
  - Same issue with protected and private
  - Comment code in GGk class
  - WaitDelays go AFTER firePropertyChanges

Practice midterm corrections

- Problem 1:
  - Show your work in a clear and understandable manner; partial credit will be given
  - Confusion of measures in the system and in the queue
  - Finding 1.a by the area under the curve

- Problem 2:
  - Trivial condition on # repairmen
  - Conditions on edge EOS→SS
  - Conditions on edges Arr→SS
  - Event graph with passing parameters across edges and issues with passing paramters
- Problem 3:
  - Discuss solution

Simple movement and detection

Simple motion: linear, uniform, and two dimensional

Cookie-cutter: sees everything within its range R and is notified the time a target enters and exits its range

Location of the entity *relative* to the sensor at time t is

$X(t) = x_0 + (t - t_0)\vec{v}$

$X(t)$ can be uniquely determined by $(x_0, t_0, \vec{v})$

where:

x_0 : is a 2-dimensional vector that represents initial position

t_0 : is the time that movement started

$\vec{v}$: is a two-dimensional velocity vector

Coordinate system: $(0,0)$ is the center of the sensor

Otherwise, if sensor is initially located at x_2 and moves with contant velocity $\vec{v}_2$:

$X(t) = (x_1 - x_2) + t(\vec{v}_1 - \vec{v}_2)$

From the DES standpoint: $X(t)$ can be determined implicitly from $(x_0, t_0, \vec{v})$

Simple detection

Goal: Find time t_d the target enters sensor range

Method: Solve

$$\|x + t\vec{v}\| = R$$

$$\|x\|^2 + 2tx \bullet \vec{v} + t^2 \|\vec{v}\|^2 = R^2$$

Solutions are given by:

$$t = -\frac{x \bullet \vec{v}}{\|\vec{v}\|^2} \pm \frac{\sqrt{\|\vec{v}\|^2 \left(R^2 - \|x\|^2\right) + \left(x \bullet \vec{v}\right)^2}}{\|\vec{v}\|^2}$$

Both roots t_d and t_e positive: Enter range after t_d and exit t_e units of time after time 0

Both roots t_d and t_e negative: Exit*ed* range t_e units of time before and entered t_d units of time prior to time 0

One real root and one negative root: Target is already in the range and will exit after a delay of the positive root

Both roots are complex: Target never gets within range of the sensor

Just one root: Targe trajectory is tangent to the sensor range

Remark: If the sensor initial position is x_s and moves with velocity $\vec{v}_s$, replace in equation above x by $x - x_s$ and $\vec{v}$ by $\vec{v} - \vec{v}_s$.

Some practice problems

1. A target is located at (10,100) (miles) and a cookie-cutter sensor is located at (40,28) with a range of 20 miles. At time 0 the target starts moving to position (60,-20) at speed 10 miles per hour.
   a) At what time does the target arrive at position (60,-20)?
   b) At what time (if any) does the sensor detect and undetect the target?
   c) Assuming the sensor does in fact detect the target, what is the location when it is detected?

2. At time 3.0 hrs a unit located at (-10,20) (km) starts moving to destination (70,60) at speed 40km/hr. A stationary sensor with range of 20km is located at (50,30).
   a) At what time (if any) will the target enter the range of the sensor?
   b) What is the (absolute) location of the target when it enters the range of the sensor?
   c) What is the minimum necessary range of the sensor that will detect our target?
   d) Suppose that the sensor's range is uniformly distributed in [10,20]. Find the probability that the target will ever be detected.
   e) What is the direction of the target's velocity vector that will make it go through the sensor's position?
   f) Give the range of directions of the target's velocity vector that will make it detectable

Solution

1.

a. We need to find the velocity vector:

$$\vec{v} = 10\frac{(60-10,-20-100)}{\sqrt{(60-10)^2+(-20-100)^2}} = 10\frac{(50,-120)}{130} = (3.85,-9.23)$$

Time target arrives at position (60,-20)= $\dfrac{\|dest-orig\|}{\|\vec{v}\|} = \dfrac{130}{10} = 13$ hrs

b. Use resolvent formula with $x = (-30,72)$ to obtain:

time to enter range t_d: 5.8

time to exit sensor range: 9.8

c. Location when target is detected is

$$x + t_d\vec{v} = (10,100) + 5.8(3.85,-9.23) = (32.2,46.5)$$

Remark: As expected we have $\|(-30,72)+t_d\vec{v}\| = 20$,

this help you check that your numbers are right